

6 I/O系统、设备

2018年11月21日 15:04

◆

◆ I/O系统的功能、模型和接口

一. I/O系统的基本功能：方便I/O、提高设备利用率、准确无误（之后省略I/O的斜杠）

1. 隐藏物理设备的细节

- 1) 不同IO设备在接收/产生数据速度、传输方向、粒度（数据细化程度）、数据表示形式、可靠性等方面都有很多差异
- 2) 各设备应配置相应设备控制器（有若干命令/参数寄存器的硬件）
- 3) 用户通过命令和参数控制外部设备执行相应操作

2. 与设备的无关性

- 1) 隐藏物理设备细节的基础上，用抽象的逻辑设备名来使用设备。如只提供读写命令和逻辑设备名，由系统自己安排打印机
- 2) 有效提高了系统的可移植性和易适应性，即不需重新编译系统即可添加新设备驱动程序。如windows即插即用IO设备

3. 提高处理机和IO设备的利用率

- 1) 尽可能让处理机和IO设备并行操作
- 2) 处理机应尽快相应IO请求，尽快运行IO设备
- 3) 尽量减少IO设备运行时处理机的干预时间

4. 对IO设备进行控制

- 1) IO软件应屏蔽差异，给高层软件 and 用户提供统一的方便操作接口
- 2) 轮询的可编程IO方式：不断测试忙闲标志
- 3) 中断的可编程IO方式：打印机，键盘等传输单位为字节/字的低速设备
- 4) 直接存储器访问方式：磁盘、光盘等传输单位为数据块的高速设备
- 5) IO通道方式：使IO操作的组织、数据的传输能无CPU干涉地独立进行

5. 确保对设备的正确共享

- 1) 共享设备：运行多进程同时访问的设备，可交叉进行读写，不影响正确性
- 2) 独占设备：应互斥访问的设备，分配给某进程后，直至用完才释放，因而需要考虑分配的安全性。如打印机、磁盘机

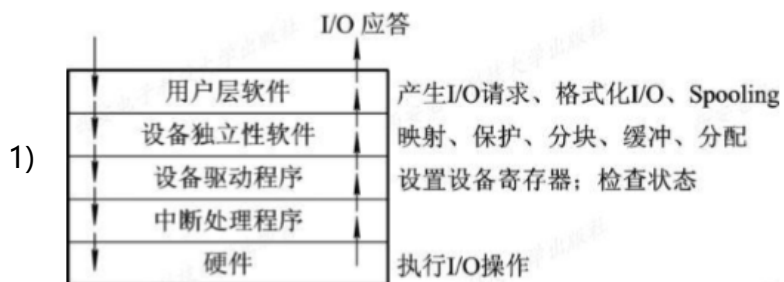
6. 错误处理

- 1) 临时性错误可通过重试来纠正
- 2) 重试多次仍无法解决将被视作持久性错误
- 3) 持久性错误一般是与设备紧密相关的，尽可能在接近硬件的层面上解决
- 4) 低层软件实在解决不了的错误才向上层报告并请求解决

二. IO系统的层次结构和模型

每一层都利用下层提供的服务，完成输入输出的某些子概念，屏蔽功能实现细节，向高层提供服务

1. IO软件的层次结构



- 2) 用户层IO软件: 实现与用户交互的接口, 用户可直接调用该层的库函数
- 3) 设备独立性软件: 实现用户程序与设备驱动器的统一接口、设备命名、设备保护、设备分配与示范等, 同时为设备管理和数据传送提供存储空间
- 4) 设备驱动程序: 与硬件直接相关, 实现操作指令, 驱动设备工作
- 5) 中断处理程序: 保持CPU环境、处理中断、恢复现场, 回到进程

2. IO系统各模块间的层次视图

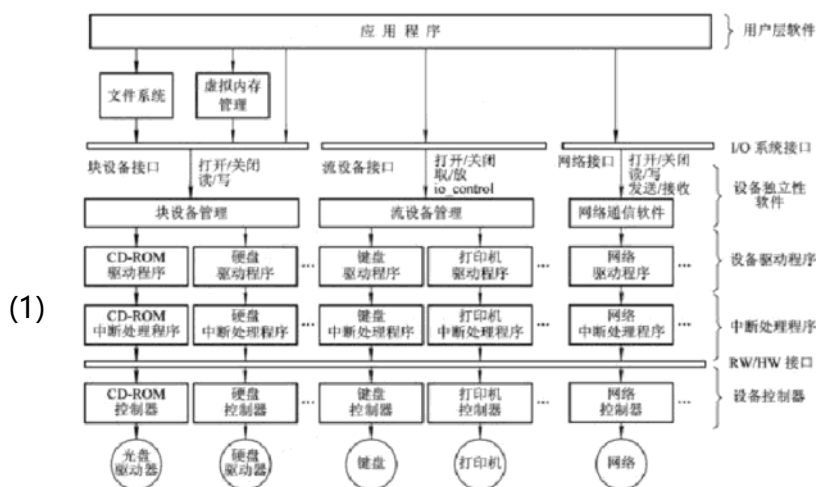


图6-2 I/O系统中各种模块之间的层次视图

1) IO系统的上下接口

- (1) IO系统接口: IO系统与上层系统 (文件系统、虚拟存储器、进程) 间的接口, 提供抽象IO命令
- (2) 软件/硬件(RW/HW)接口: 上层是中断处理、设备驱动, 下层是设备控制器 (CD-ROM、硬盘、键盘、打印机、网络等的控制器)

2) IO系统的分层

- (1) 中断处理程序: 最底层, 与硬件直接交互, 当IO设备发出中断请求后, 做出初步处理便转向中断处理程序。保存环境、转入处理、恢复环境、返回断点
- (2) 设备驱动程序: 次底层, 是进程和设备控制器之间的通信程序。把上层抽象IO请求转换为具体命令和参数, 并装入设备控制器的寄存器中, 或相反。每次增加新设备到系统, 都要安装新驱动
- (3) 设备独立性软件/设备无关性软件: 独立于具体使用物理设备的IO软件, 提高了系统的可适应性和可扩展性。包括设备命名、设备分配、数据缓冲、数据高速缓冲等软件

三. IO系统接口

1. 块设备接口：块设备管理程序与高层间的接口，控制其输入输出
 - 1) 块设备：以数据块为存取传输单位的设备，如磁盘
 - (1) 传输效率较高，数m到数十m每秒，可寻址，常用DMA方式IO
 - 2) 隐蔽磁盘二维结构，将扇区从0到n-1编号，用磁道号和扇区号线性表示扇区地址
 - 3) 将抽象命令映射为低层操作
 - 4) 如缺页中断后，IO系统通过块设备接口从磁盘将所缺页调入内存
2. 流设备接口：流设备管理程序与高层间的接口，控制其输入输出
 - 1) 流设备/字符设备：以字符为单位，如键盘、打印机
 - (1) 传输效率低，数到数千字节每秒，不可寻址，常用中断驱动方式IO
 - 2) get/put操作：在字符缓冲区，顺序存取字符
 - 3) in-control指令，包含了许多参数，对应具体设备相关的特定功能
 - 4) 基本都是独占设备，需要互斥共享，使用前后需要打开/关闭
3. 网络通信接口：通过网络与网络上其他计算机通信或上网浏览



◆ IO设备和设备控制器

1. 执行IO操作的机械部分就是一般IO设备，执行控制IO电子部件则成为设备控制器或适配器adapter
2. 微、小型机中的控制器一般是插入扩展槽中的印刷电路卡，因此又称控制卡、接口卡、网卡
3. 中、大型机一般配置IO通道或IO处理机

一. IO设备

1. IO设备类型

1) 按使用特性分类

- (1) 存储设备/外存/后备存储器/辅助存储器，是存储信息的主要设备。
存取速度较内存慢，但容量大得多，价格便宜
- (2) 输入设备、输出设备、交互式设备
 - i. 输入设备接收外部信息，如键盘、鼠标、扫描仪、视频摄像、各类传感器等
 - ii. 输出设备将信息送向外部，如打印机、绘图仪、显示器、音响等
 - iii. 交互式设备集成上述两类

2) 按传输速率分类

- (1) 低速设备：传输速率仅每秒几字节至数百字节。键盘、鼠标器、语音的输入和输出等
- (2) 中速设备：传输速率在每秒数千字节至数十万字节。行式打印机、激光打印机等
- (3) 高速设备：传输速率在数百千字节至千兆字节。磁带机、磁盘机、光盘机等

3) 按信息交换的单位分类：块设备、字符设备

4) 按共享属性分类：独占、共享、虚拟设备

2. 设备与控制器之间的接口：设备通常不直接与CPU进行通信，只跟控制器通信

- 1) 数据信号线：在设备及其控制器之间传送数据信号
- 2) 控制信号线：控制器向IO设备发送控制信号时的通路
- 3) 状态信号线：传送设备当前状态信号的线

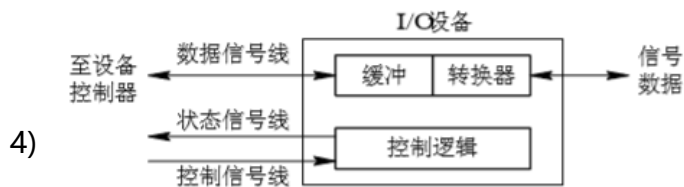


图 5-1 设备与控制器间的接口

二. 设备控制器：设备与CPU之间的接口

1. 设备控制器基本功能

- 1) 接收和识别命令：用控制寄存器和命令译码器，存放接收CPU的命令和参数，并进行译码
- 2) 数据交换：用数据寄存器和信号线实现 CPU 与控制器之间、控制器与设备之间的数据交换
- 3) 标识和报告设备的状态：用状态寄存器供 CPU 了解设备状态
- 4) 地址识别：用地址译码器识别每个设备对应的唯一地址
- 5) 数据缓冲：用缓冲器减缓设备与CPU或内存的速率不匹配问题
- 6) 差错控制：将差错检测码置位，并报告CPU将本次传送来的数据作废，并重新进行一次传送。保证数据输入的正确性

2. 设备控制器的组成

- 1) 设备控制器与处理机的接口：数据线、地址线和控制线
 - (1) 数据线通常与数据寄存器或控制/状态寄存器相连接
- 2) 设备控制器与设备的接口
 - (1) 每个接口中都存在数据、控制 and 状态三种类型的信号
- 3) I/O 逻辑：实现对设备的控制
 - (1) 处理机通过一组控制线利用该逻辑向控制器发送 I/O 命令
 - (2) I/O 逻辑对收到的命令进行译码，再对所选设备进行控制
 - (3) 每当 CPU 要启动一个设备时，一方面将启动命令发送给控制器；另一方面又同时通过地址线把地址发送给控制器，由 I/O 逻辑对收到的地址进行译码

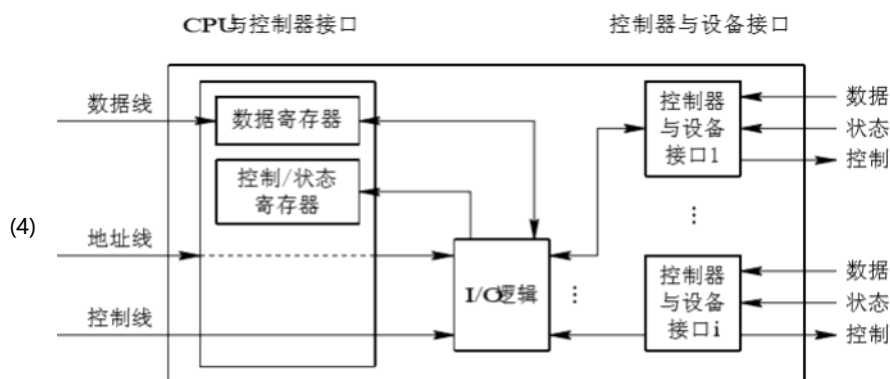


图 5-2 设备控制器的组成

三. 内存映像IO：将抽象IO命令转换出的命令/参数装入控制器的寄存器

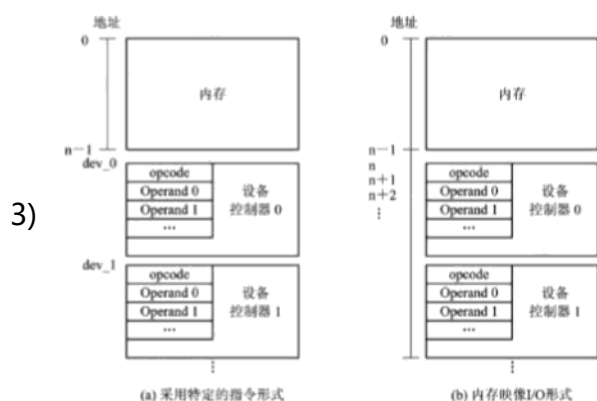
1. 利用特定的IO指令

- 1) 如Store cpu-reg k将cpu的reg寄存器的数据存入内存中的k单元
- 2) 访问设备可能需要不同的指令，主要指单元记法不同

2. 内存映像IO

- 1) 编址上不再区分内存单元地址和设备控制器中的寄存器地址，都采用k

2) 当k大于等于n时被认为是寄存器址（因为内存地址从0计，最大值为n）



四. IO通道

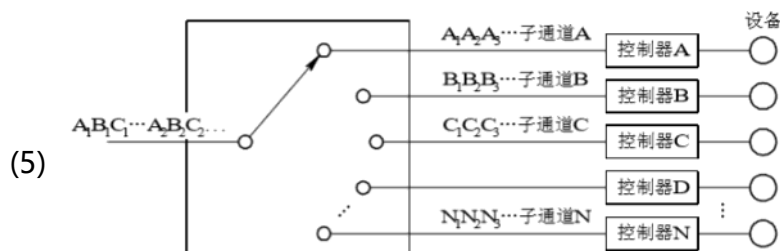
1. IO通道设备的引入

- 1) 目的是使IO操作不仅在数据传送，而且在IO操作组织、管理、结束处理都尽量独立于CPU
- 2) **IO通道可视作特殊的处理机**，有执行IO指令的能力，特殊在：
 - (1) 指令类型单一，只执行与IO操作有关的指令
 - (2) 没有自己的内存，只借用主机内存，即与CPU共享内存

2. 通道类型

1) 字节多路通道(Byte Multiplexor Channel)

- (1) 按字节交叉方式工作
- (2) 通常都有许多非分配性子通道，数量从几十到数百个，每个子通道连接一台I/O设备，并控制其I/O操作
- (3) 子通道按时间片轮转方式共享主通道
- (4) 不适于连接高速设备，会容易丢失信息



2) 数组选择通道(Block Selector Channel)

- (1) 按数组方式传送，只有一个分配性子通道，一段时间内只能执行一道通道程序，控制一台设备，直至被释放才能供下一设备使用

3) 数组多路通道(Block Multiplexor Channel)

- (1) 两者结合，多个非分配型字通道，按数组方式传送
- (2) 高传输速率和高通道利用率，广泛用于多台高中速外围设备系统

3. 瓶颈问题：通道价格昂贵，数量势必较少，进而造成系统吞吐量下降

- 1) 单通路方式：如图，设备1、2、3任一个使用时，4就不能使用



图 5-4 单通路 I/O 系统

2) 多通路方式：增加设备到主机间的通路而不增加通道

- (1) 即把一个设备连上多个控制器
- (2) 不仅解决了瓶颈问题，还提高了系统可靠性，个别通道或控制器故障不会导致设备和存储器间无通路

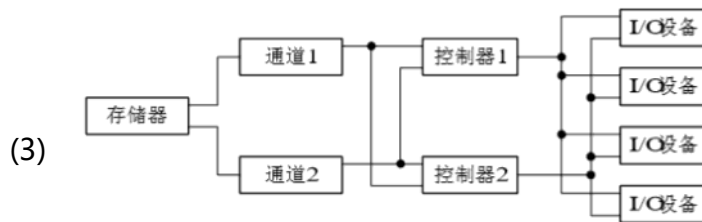


图 5-5 多通路 I/O 系统

五. 总线系统：实现 CPU、存储器以及各种 I/O 设备之间的联系的系统

1. ISA 和 EISA 总线

- 1) ISA(Industry Standard Architecture)总线：1984 年推出的微机的总线结构。其带宽为 8 位，最高传输速率为 2 Mb/s。之后不久又推出了 16 位的(EISA)总线，其最高传输速率为 8 Mb/s，后又升至 16 Mb/s，能连接 12 台设备
- 2) EISA(Extended ISA)总线：20 世纪 80 年代末期，为满足带宽和传输速率的要求，开发出扩展 ISA(EISA)总线，其带宽为 32 位，总线的传输速率高达 32 Mb/s，同样可以连接 12 台外部设备

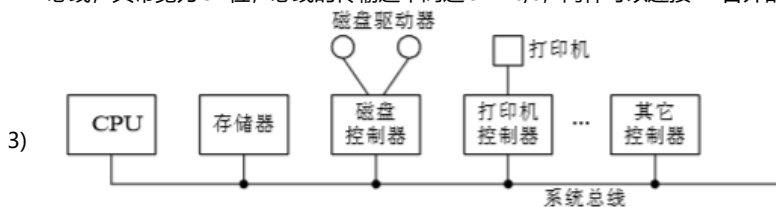


图 5-6 总线型 I/O 系统结构

2. 局部总线(Local Bus)

- (1) 多媒体技术的兴起，特别是全运动视频处理、高保真音响、高速 LAN，以及高质量图形处理等技术，都要求总线具有更高的传输速率，于是，局部总线便应运而生
- (2) 局部总线：将多媒体卡、高速 LAN 网卡、高性能图形板等，从 ISA 总线上卸下来，再通过局部总线控制器直接接到 CPU 总线上，使之与高速 CPU 总线相匹配，而打印机、FAX/Modem、CDROM 等仍挂在 ISA 总线上。在局部总线中较有影响的是 VESA 总线和 PCI 总线

1) VESA(Video Electronic Standard Association)总线

- (1) 其设计思想是以低价位迅速占领市场。带宽为 32 位，最高传输速率为 132 Mb/s。它在 20 世纪 90 年代初被推出时，广泛应用于 486 微机中。仍存在较严重的缺点，如，能连接的设备数仅为 2~4 台，在控制器中无缓冲，难于适应处理器速度的不断提高，也不能支持后来出现的 Pentium 微机

2) PCI(Peripheral Component Interface)总线

- (1) 随着 Pentium 系列芯片的推出，Intel 公司颁布了 PCI 总线的 V1.0 和 V2.1 规范，后者支持 64 位系统。PCI 在 CPU 和外设间插入一复杂的管理层，协调数据传输和提供一致的接口。在管理层中配有数据缓冲，通过该缓冲可将线路的驱动能力放大，使 PCI 最多能支持 10 种外设，并使高时钟频率的 CPU 能很好地运行，最大传输速率可达 132 Mb/s。PCI 既可连接 ISA、EISA 等传统型总线，又可支持 Pentium 的 64 位系统，是基于奔腾等新一代微处理器而发展的总线

- i.
- ii.
- iii.
- iv.
- v. -----我是底线-----